

SINGLE POSITION NON-INVASIVE CALIBRATION TECHNIQUE FOR RESPIRATION MONITORING APPARATUS

Patent number: JP63502563T
Publication date: 1988-09-29
Inventor:
Applicant:
Classification:
- international: A61B5/08
- european:
Application number: JP19870501323 19870121
Priority number(s): US19860820604 19860121

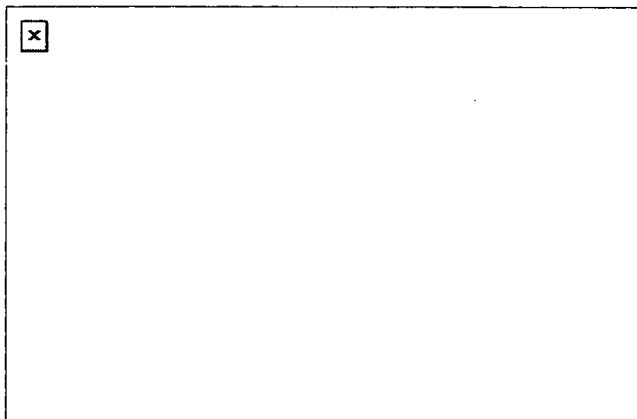
Also published as:

WO8704332 (A1)
EP0256115 (A1)
EP0256115 (A4)
EP0256115 (B1)

Abstract not available for JP63502563T

Abstract of correspondent: **WO8704332**

Method for non-invasively measuring respiration volume whereby weighting factors are non-invasively determined by (a) totaling delta values over a baseline period of substantially steady state breathing for obtaining a rib cage signal, (b) totaling delta values over a baseline period of substantially steady state breathing for obtaining an abdominal signal, (c) dividing the average variability of the mean of the total of the delta values for one of either the rib cage or abdomen signals by the average variability of the mean of the total of the delta values for the other of either the rib cage or abdomen signals; and (d) multiplying the other signal by the quotient derived from step (c).



Data supplied from the **esp@cenet** database - Worldwide

⑨公表 昭和63年(1988)9月29日

⑥Int.Cl.⁴
A 61 B 5/08

識別記号

庁内整理番号
7916-4C

審査請求 未請求
予備審査請求 未請求

部門(区分) 1(2)

(全 31 頁)

④発明の名称 呼吸モニター装置用の単一姿勢における非侵襲性校正技法

⑪特 願 昭62-501323

⑫出 願 昭62(1987)1月21日

⑬翻訳文提出日 昭62(1987)9月21日

⑭国際出 願 PCT/US87/00217

⑮国際公開番号 WO87/04332

⑯国際公開日 昭62(1987)7月30日

優先権主張 ⑰1986年1月21日⑱米国(US)⑲820,604

⑳発 明 者 ワトソン・ヘルマン 米国, フロリダ 33139-1432 マイアミ, ウェスト・アベニュー 1842

㉑出 願 人 レスビトレイス・コーポレーション 米国, フロリダ 33139-1432 マイアミ, ウェスト・アベニュー 1842

㉒代 理 人 弁理士 野河 信太郎

㉓指 定 国 A T(広域特許), A U, B E(広域特許), B R, C H(広域特許), D E(広域特許), D K, F I, F R(広域特許), G B(広域特許), I T(広域特許), J P, K R, L U(広域特許), N L(広域特許), N O, S E(広域特許), S U

特許請求の範囲

1.呼吸量に対する胸部部番号を表示する胸部部量に 대응する信号を提供し、呼吸量に対する腹部の番号を表示する腹部量に 対応する信号を提供し、前記胸部部及び腹部信号の少なくとも 一方に、呼吸量に対する前記胸部部及び腹部の相対的番号を反 映する所定の重み係数を乗じ、呼吸量に比例する信号を提供す るために前記の重みを付けた信号を加算する方式の被験者の呼 吸量を非侵襲的に測定する技法において、

(a)実質的に定常状態にある呼吸のベースライン期間にわた る前記胸部部信号用のデルタ値を合計し、

(b)実質的に定常状態にある呼吸のベースライン期間にわた る前記腹部信号用のデルタ値を合計し、

(c)前記胸部部または腹部信号の一方の前記デルタ値の合計 の平均値の平均変化性を、前記胸部部又は腹部信号の他 方の前記デルタ値の合計の平均値の平均変化性によって 除し、そして

(d)前記他方の信号に、ステップ(c)から得られた商に等しい 重み係数を乗ずる

ことによって前記重み係数を非侵襲的に決定することを特徴 とする技法。

明 細 書

呼吸モニター装置用の単一姿勢における非侵襲性校正技法

技術分野

この発明は呼吸量を測定するための方法および装置に関し、 詳しくは胸部部や腹部のような胴体部の複数箇所からの番号 (contribution)を個々に測定し合計することによって呼吸量 を測定する方法および装置に関する。さらに詳しくは、この発 明は胴体部からの番号を示しそれによって信号の合計が呼吸量 に比例する、重み信号のための校正技法に係わるものである。

背景技術

呼吸量測定のための方法および装置と題された1982年1 月5日付けの米国特許第4,308,872号は、この明細書に 引例としてそのまま挿入されるが、その特許は呼吸量を定量的 に測定するための方法および装置を開示している。上記特許に 開示されている方法は、胸部部および腹部の周りに伸張性のあ る第1および第2の伝導体を巻き付け、呼吸の間に伝導体のイン グクタン스를個々にそして同時に測定し、呼吸量に対し胸部部 および腹部の異なる番号を反映させるため測定されたイン グクタン스에重みを与え、そして実際の呼吸量を得るために、重 んが与えられ測定されたイングクタン스를合計する方法からな る。上述した特許に開示された技法の実施は、胸部部および腹 部伝導体によって測定されたイングクタン스에重みを与えるこ とまたは校正することを必要としている。校正を実施するため

が必要である。

$$V = K \cdot RC + L \cdot AB \quad \text{【方程式 A】}$$

ここにVは全体の呼吸量、RCは胸郭部伝導体で測定された呼吸量に対する胸郭部寄与、そしてABは腹部で測定された腹部寄与である。米国特許第4,308,872号は、重み係数KおよびLとしての数値を決定する特別な方法を開示している。

その開示によれば、校正手続き中は肺活量計が使用される。起立姿勢のような第1の姿勢にある患者について、併発する一組の読み取りが肺活量計、胸郭部伝導体および腹部伝導体の出力から記録される。記録は仰向け姿勢のような第2の姿勢にある患者に対しても繰り返される。この時点で方程式Aを満足するV、RCおよびAB用の二組の数値がある。したがって2組の未知数、定数KおよびLを有する2組の式が得られる。これらの式により、重み係数KおよびLは連立式を解く未知の計算方法によって決定される。

$$K = \frac{AB_1 \cdot V_2 - AB_2 \cdot V_1}{RC_1 \cdot AB_2 - RC_2 \cdot AB_1} \quad \text{【方程式 B】}$$

$$L = \frac{RC_1 \cdot V_2 - V_1 \cdot RC_2}{AB_1 \cdot RC_2 - AB_2 \cdot RC_1} \quad \text{【方程式 C】}$$

記録された数値に依存する、式BおよびCの分母は、ゼロに

およびAB/Vであるグラフ上にプロットされ、そしてこれらの点を通しておおよその線が引かれる。この線はおおよその目視によって引かれるが、最小二乗法によって決定されることが好ましい。線はそれからX軸およびY軸に達するまで延長される。XおよびY切片の逆数は、重み係数KおよびLを決定する。すなわちRC/V軸の切片の逆数は胸郭部用の重み係数Kを決定し、AB/V軸の切片の逆数は腹部用の重み係数Lを決定する。上述した計算の総ては、重み係数KおよびLのための数値を演算して算出するマイクロプロセッサまたは他のデータプロセッサによって行われることが望ましい。米国特許第4,308,872号および第4,373,534号に開示されている方法および装置の欠点は、換気における二つの異なる位置すなわち被験者に対し離れた二箇所に取り付けられる、腹部および胸郭部伝導体から、肺活量計または他の呼吸量測定装置からとの一組のデータ点または数値を必要とすることである。被験者の生理的條件が、ある姿勢から他の姿勢への移動を阻止または規制する場合、しかしながら胸郭部および腹部伝導体の校正寄与は既知の方法に従って容易に実行できない。これらの先行技術の方法および装置の他の欠点は、新生児の測定のようにある適用においては肺活量計のような独立した呼吸量測定装置を使用して装置を校正することは実用的ではないことである。例えば上記の校正方法は肺活量計または他の同等装置に接続する通気路を必要とするので、測定手続きを実行するためには著しい時間が必要とされる。これは時間が貴重である新しく生まれた看護婦職員にとってし

合、KおよびLのために得られた数値は明らかに不正確となり、したがってそのような重み係数を基にしたいかなる測定も益んだものとなる。よって式BおよびCの分母がゼロに接近するかまたはゼロに等しくなるときは常に新しい一組の読み取りが取り入れなければならない、それゆえ校正のために要する時間が増加していた。

呼吸量測定装置を校正するための方法および装置と題された1983年2月15日付け米国特許第4,373,534号におけるその内容もここにその総てを引用として挿入されるが、その特許には重み係数KおよびLを決定するための、図示を基にした技法である代わりの方法および装置が開示されている。米国特許第4,308,872の連立式に従う技法におけるごとく、呼吸量を個々に測定するための肺活量計または他の装置が、校正の手続きの間使用される。第1姿勢の被験者に対して肺活量計、胸郭部伝導体、腹部伝導体からの読み取りは、複数の呼吸の間隔もしくは少なくとも全部で3回、同時に記録される。この記録は第2姿勢の被験者に対しても繰り返される。各呼吸について胸郭部および腹部の読み取りは肺活量計の読み取りによって分割される。すなわち、RC/V値およびAB/V値は、各呼吸について得られ、ここにVは肺活量計によって測定された呼吸量であり、RCは未校正の胸郭部伝導体からの胸郭部読み取りであり、ABは未校正の腹部伝導体からの腹部読み取りである。各呼吸についての点(RC/V、AB/V)は次に、中心線がRC/Vお

よびAB/Vであるグラフ上にプロットされ、そしてこれらの点を通しておおよその線が引かれる。この線はおおよその目視によって引かれるが、最小二乗法によって決定されることが好ましい。線はそれからX軸およびY軸に達するまで延長される。XおよびY切片の逆数は、重み係数KおよびLを決定する。すなわちRC/V軸の切片の逆数は胸郭部用の重み係数Kを決定し、AB/V軸の切片の逆数は腹部用の重み係数Lを決定する。上述した計算の総ては、重み係数KおよびLのための数値を演算して算出するマイクロプロセッサまたは他のデータプロセッサによって行われることが望ましい。米国特許第4,308,872号および第4,373,534号に開示されている方法および装置の欠点は、換気における二つの異なる位置すなわち被験者に対し離れた二箇所に取り付けられる、腹部および胸郭部伝導体から、肺活量計または他の呼吸量測定装置からとの一組のデータ点または数値を必要とすることである。被験者の生理的條件が、ある姿勢から他の姿勢への移動を阻止または規制する場合、しかしながら胸郭部および腹部伝導体の校正寄与は既知の方法に従って容易に実行できない。これらの先行技術の方法および装置の他の欠点は、新生児の測定のようにある適用においては肺活量計のような独立した呼吸量測定装置を使用して装置を校正することは実用的ではないことである。例えば上記の校正方法は肺活量計または他の同等装置に接続する通気路を必要とするので、測定手続きを実行するためには著しい時間が必要とされる。これは時間が貴重である新しく生まれた看護婦職員にとってし

ばしば歓迎されなかった。

これらの方法および装置におけるさらに他の問題は、呼吸器の装置における空気の移動は常に胸郭部と肺活量計との間か、または腹部と肺活量計との間を移動するという前提に依存することである。しかし実際は通常の呼吸中に、胸郭部および腹部の間で空気の移動が存在する。この空気のRC-AB交換は、連続して発生し、呼吸における量を変化させるベンデルフト(pendelfut)である。既知の方法はベンデルフトとしての一定量を加えることができなかった。

他の既知の定量的校正方法が、ワトソン他等の「呼吸器疾患容積計の単一姿勢間呼吸校正」、(Aner. Rev. of Respiratory Diseases, Vol. 129, p. 1256 (1984))に記載されている。この方法によると、単一の姿勢または状態にある患者に対し肺活量計(SP)、胸郭部伝導体(RC)および腹部伝導体(AB)からの読み取りは、好ましくは少なくとも一回の完全呼吸の、所定期間同時に記録される。次いで曲線(SP, RC)および曲線(SP, AB)は記録されたデータからプロットされ、得られる各曲線はその曲線の始点および終点とを結ぶ直線によって閉じられる。次いで生じるループ面積が積分によって演算される。次に肺活量計(SP)から、胸郭部(RC)および腹部(AB)伝導体から同時に記録されたいずれかの選択されたデータ点を使用し、重み係数(胸郭部比例増幅器および/または腹部比例増幅器用)が決定される。この校正技法もまた、呼吸量を口で測定する肺活量計または他の装置を必要とし、ある

図面の簡単な説明

図面において、いくつかの図面を通して同じ参照番号であるところは同じ要素であることを示す。

第1図は非侵襲性呼吸量測定装置の一部の線図である。

第2図は非侵襲性呼吸量測定用の完成装置のブロックダイアグラムである。

第3図は胸郭部および腹部信号用のデルタ値を示すグラフ図である。

この発明を実施するための最も好ましい実施態様

さて、図面を参照すると、米国特許第4,306,872号に開示されたタイプの呼吸量測定装置が、第1及び第2図に示されている。伸縮性および伝導性を有する2つのループ12、14は、任意の適当な方法で弾性チューブ16、18にそれぞれ設置され、伝導体12と14は被験者の胸郭部と腹部に各々周設される。被験者10が呼吸すると、弾性チューブ16、18と伝導性ループ12、14は膨張・収縮し、その結果としてループのインダクタンスの変化を生ずる。各ループのインダクタンスは、比例信号に変換され、それらの信号は校正されて加算され、干渉量(tidal volume)を表示する信号を提供する。干渉量に対する胸郭部と腹部の相対的な寄与は、被験者が交替したり、同じ一人の被験者であっても例えば立ったり仰向けになったりその他の異なる姿勢をとるなどによって変化するもので、胸郭部と腹部の伝導体12と14からの信号の校正がそれぞれ必要と

マイクロプロセッサ34が、増幅器からの信号を加算するために使用される場合には、加算増幅器28は除去される。

この発明は、胸郭部と腹部からの信号を校正し、それらを加算することによって干渉量を表示する信号を得る改良された方法のためのものである。上記のように、種々の校正技術は当業者に公知であるが、全知技術の全ては欠点を有している。

この発明の校正技術によると、方程式Aから次のことが知られる。

$$V = (K \times RC) + (L \times AB)$$

ここでVは合計呼吸量または干渉量、RCは胸郭部伝導体によって測定される胸郭部の呼吸量に対する寄与であり、ABは腹部伝導体によって測定される腹部の呼吸量に対する寄与である。KとLはそれぞれ胸郭部と腹部に対する校正係数である。この関係を他の方法で表現すると、

$$V = M \times [(Z \times RC) + AB] \quad \text{【方程式D】}$$

ここでM×ZはKに等しく、MはLに等しい。方程式Dは校正係数を比例係数Zと増幅係数Mとに分離していることは明らかである。この手順を用いることによって、校正は2段階の処理として見なすことができる。第1の段階は方程式Dを満足させる正しい比例係数Zを決定することである。つまり

$$V = (Z \times RC) + AB \quad \text{【方程式E】}$$

換言すれば、比例係数Zは、胸郭部(RC)と腹部(AB)との、干渉量(V)に対する正しい相対的な寄与を定義している。古典的には、比例係数Zの決定は、被験者が気道を閉じて

適当な相対的な寄与に反映させるためにこれらの信号を校正すると共に、これらの信号を加算して干渉量を表示する信号を提供する適当な装置は、当業者にとって公知である。このような装置の1つは、米国特許第4,306,872号に開示されている。更に他の適当な装置は、フロリダ州マイアミビーチのニムズ(株)によってリズビグラフTMという商名で市販されている。そのような装置は、一般に第2図のブロックダイアグラムの形で示され、そこでは、ブロック20、22が、胸郭部および腹部の伝導体12、14のインダクタンスを、連続処理される適当な比例電気信号に変換する適切な回路をそれぞれ表している。増幅増幅器(scaling amplifiers)24、26は、胸郭部と腹部からの信号をそれぞれ校正して、胸郭部と腹部の相対的な寄与を干渉量に反映させる回路を表現している。増幅増幅器24、26が適当に校正されると、その結果生ずる信号は、第2図の加算増幅器(summing amplifier)28によって加算され、干渉量を表示する信号が出力される。加算増幅器28からの出力信号は、2つの増幅増幅器24と28からの出力信号と同様に、グラフィックレコーダ30又はデジタルボルトメータ32に表示される。増幅増幅器24、26からの信号を加算し、そして/または、診断上の目的のためにこれらの信号を更に処理する装置に、マイクロプロセッサ34が選択的に組み入れられるが、すべては、当業者にとって公知の技術によるものである。もし、

呼吸し、つまり口において動き量が全くないようにし、それによってV=0となる等容の(isovolune)校正技術によって行われる。つまり、等容処理(isovolune maneuver)の間、空気は口を通じて全く逃げることはないで、量の動きは胸郭部と腹部との区画間、つまりペンデルフト(pendulift)だけである。このような条件のもとで、方程式Dは次のようになる。

$$0 = (Z \times RC) + AB \quad \text{【方程式F】}$$

あるいは、

$$Z = -AB/RC \quad \text{【方程式G】}$$

そこで、等容処理期間における胸郭部と腹部の伝導体からの読み取りを記録することによって、比例係数Zは方程式Gから決定することができる。Zが決定されると、量(Z×RC)+ABは、胸郭部と腹部の伝導体の記録値から任意の時刻に校正される。方程式Eから(Z×RC)+ABは常に干渉量Vに比例することが知られるので、それらの量の決定は、低価な診断器具を提供する。たとえば、当業者であれば認識していることであるが、この量から閉塞性の中核呼吸を診断することができ、RCは干渉量のVの1%として校正され、そして相対的な干渉量Vにおける増減が決定される。

このような方法にともなう困難な点は、等容処理が閉じた気道に対して呼吸を要求することであり、新生児や危険な患者管理に適用するような場合には必ずしも実用的ではないということである。

一方、もし比例係数Z、従って量(Z×RC)+ABが、等

ねばならず、そうでない場合には2つの未知数 Δ と Δ と干渉量 V を伴う単一の方程式が存在するという事は方程式Eから明らかである。

この発明によれば、この問題が次のように解決される。

胸郭部(RC)と腹部(AB)の信号は、最初のベースライン(base line)期間において数多くの呼吸に対して記録される。例えば、250回の呼吸が、10分間隔で、安静な呼吸期間において測定される。実際には、ベースライン期間の呼吸は、安静状態でさえあれば安静である必要はない。たとえば、ベースラインは、10分の実行期間において記録される呼吸から引出すことができる。このベースライン期間の間に胸郭部(RC)と腹部(AB)の低周波からの未校正の信号が記録される。第3図を参照すると、これらの信号の各々に対して、2つの値つまり呼吸「デルタ」が呼吸毎に校正され、「 Δ 」は吸息の始めと終わりの信号間の差であり、もう一つの「 Δ 」は呼息の始めと終わりの間の差である。これらのデルタ値は、各信号に対して個別に合計される。ベースライン期間における250回の呼吸を仮定すると、各呼吸に対して2つのデルタの値が存在するので、各信号の合計は500のデルタ値を加えることによって演算される。上記のデルタ値が好ましい場合には、ベースライン期間中に行われる未校正の胸郭部及び腹部信号の各呼吸に対して相対的な増幅度を表すパラメータを提供するためにそのデルタ値が採用されることは、昇格されるべきである。

となる。

ここで平方偏差(Var)は(SD)²に等しい。方程式Jは次のようにも表される。

$$Z^2 \times \text{Var}(RC) = \text{Var}(AB) \quad \text{【方程式K】}$$

両辺の平方根を取ることによって、

$$Z \times \text{SD}(RC) = \text{SD}(AB) \quad \text{【方程式L】}$$

そしてこれは、

$$Z = \text{SD}(AB) / \text{SD}(RC) \quad \text{【方程式M】}$$

となる。

胸郭部と腹部の信号の平均の標準偏差は、これらの信号の平均変化度を表すということは評価されるべきである。従って、これらの信号の平均的な変化度の表示を提供する任意の解析は、標準偏差を計算する代わりに用いられる。

換言するならば、方程式Eの解に対する比例係数Zは、ベースライン期間における一定の干渉量の呼吸を仮定することにより、等量処置に類似した方法で、しかし気道をふさぐという要求なしに、ABとRCの標準偏差の比から算出し得るということとは、方程式Mから明らかであろう。

もちろんベースライン期間において肺活量計で肺活量測定を採取することは、この発明による、つまり、非侵襲的に校正するという校正技術の目的を覆すものである。

しかしながら、侵襲することなしに実際の干渉量を記録するこの発明の校正技術を実施するために要求されるベースライン期間中の一定干渉量の仮定は、RCとABの成分の未校正の和

味する。

もし、実際の干渉量が、ベースライン期間中に肺活量測定によって記録され、そして干渉量に対するデルタが累計され、さらにその平均が決定される場合には、次の関係が適用される。

$$\text{平均SP} = \text{平均RC} + \text{平均AB} \quad \text{【方程式H】}$$

ここでSPは例えば肺活量測定によって決定された実際の干渉量である。方程式Hにおける平均値は未校正信号から引き出されるので、もし方程式Hにおける左辺が右辺に等しい場合には、校正係数が要求される。

干渉量(SP)、胸郭部(RC)そして腹部(AB)信号に対する平均値の標準偏差(SD)が算出される。ここで、これらは、それぞれSD(SP)、SD(RC)、そしてSD(AB)として表される。もし、干渉量(SP)が、ベースライン期間中に記録される全ての呼吸に対して一定であれば、方程式Hは、やはり演算されることが可能で、干渉量の標準偏差SD(V)は0である。

この状態は方程式Fに類似しており、上述のようにV=0である等容処置に適応される。特に、一定の干渉量の標準偏差SD(V)これも0であるがこれを考慮することによって、正規の呼吸期間中に生じるペンデルフトは、方程式Pの等容状態に類似した状況を生み出し、両辺の平方偏差(Var)を取ると、

$$\text{Var}(Z \times RC) = \text{Var}(AB) \quad \text{【方程式J】}$$

の標準偏差1.5の外側の値を除去することによって、ベースライン期間中に算出されるABとRCの500個のデルタ値から見当違いの点を除去することにより満足させられる。これらのデルタ値が排除されると、RCとABのために残されたデルタ値が別個に合計され、その平均が決定され、そして、その平均の標準偏差が算出される。従って比例係数Zは方程式Mから算出される。

比例係数Zが既知である時、量(Z×RC)+ABは、実際の干渉量に常に比例するであろう(方程式E参照)。この量は、第2図の装置によって出力される胸郭部と腹部のリアルタイム信号から、リアルタイム基準において連続的にモニターされる。この量は、平均RCと平均ABが、それぞれ、ベースライン時に出力される未校正のRCとABのデルタ値の和の平均であり、見当違いの点を除いたものである場合に、(Z×平均RC)+平均ABの1%として表わされることが好ましい。これは、閉塞性の中気腫呼吸や、ハイポアペネアス(hypoapnea)や、干渉量の变化のような大半の診断作業に対して十分なものであり、従って広範な種々の病気における診断用具として重要である。

第2図を参照して、増幅増幅器が最初に単一のゲインに設定されているものと仮定する場合、比例係数Zが決定されると、胸郭部のための増幅増幅器24はZに調整される。方程式Eから、加算増幅器28の出力における増幅増幅器からの信号の和は、干渉量に比例するということがわかる。これによって校正手順は完了する。

START THE DATA COLLECTION

```

XRA A
STA VALFLG
STA POSI NOT IN SINGLE POSITION
LXI H,0
SHLD NBERS IN VALUES IN XVALUES & YVALUES

LXI H,0
SHLD TIME
LXI H,0
SHLD MAXCNT

LOA BSEFLG
IF (.A,IS,ZERO) NOT COLLECTING BASELINE
INIT THE TRANSDUCERS
MVI A,10
CALL SYST
THIS RESETS CALS TO 1 AND 1.50 AFTER THIS IS EXECUTED, ALL OTHER
CALIBRATIONS ARE NULLIFIED
XRA A
STA CALTYPE
STA LASTCAL (RIGHT NOW UNCALIBRATED)
STA OODDBASE (NO LONGER HAS GOOD BASELINE)

CALL LPON
PRM RES,RESPT
CALL LPOFF
ENDIF

CALL SETUP
CALL SRTINIT (INIT RT. PLOT)
LOOP
CALL SRTPLT

LOA BFLG
IF (.A,IS,ZERO)
UPDATE B DELTAS AND ELAPSED TIME
MVI A,3
STA OPCODE
CALL ANSOUT
ELSE
LOA SECFLG
IF (.A,IS,ZERO)
MVI A,3
STA OPCODE

```

```

ENDIF
ENDIF

LHLD MAXCNT
LXI D,3
ORA A
DSEC 0
IF (PSU,IS,CARRY)

LOA UTFLG
IF (.A,IS,ZERO)
MVI A,1
STA OPCODE (NOT ENOUGH DELTAS)
CALL ANSOUT
MVI A,0
STA OPCODE (PRINT MENU ON SCREEN)
CALL ANSOUT
ENDIF
STC
RET
ENDIF

```

THERE ARE 2*MAXCNT - 1) FLOATING PT VALUES IN XVALUES AND YVALUES FOR THIS RUN

```

LHLD MAXCNT
SHLD NBERS (TOTAL B STORED)
CALL DGCAL
RETURN WITH CARRY SET IF BAD FOR ANY REASON

```

```

ORA A
MVI A,0
STA OPCODE (PRINT MENU ON SCREEN)
CALL ANSOUT
RET

```

```

ISOCALC:
LOA URAPF
IF (.A,IS,ZERO) THEN 400 PTS AND RIPT PTS TO 1ST DATA PT

LXI H,400
SHLD CNT
LHLD RIPT
SHLD ROPT
ELSE
TIME HOLDS B OF DATA POINTS AND 1ST PT AT OFFSET = 0
LHLD TIME
SHLD CNT

```

```

STA FLSHFLG
IF (.A,IS,ZERO)
LXI H,TOPLFT*53
LOA BSEFLG
IF (.A,IS,ZERO)
LOA ISDFLG
IF (.A,IS,ZERO)
LXI D,CLEARNESS
ELSE
LXI D,ISQNESS
ENDIF
ELSE
LXI D,OSSENESS
ENDIF
CALL LABEL
MVI A,2
STA FLSHFLG
ELSE
LXI H,TOPLFT*53
LXI D,BLANKS
CALL LABEL
ENDIF

```

```

ENDIF
ENDIF

LOA GFLG (OFF THEN CLOCK)
IF (.A,IS,ZERO)
EXITIF(.A,IS,ZERO)
ENDLOOP

```

NOW DATA COLLECTION PHASE HAS BEEN COMPLETED
MAXCNT HOLDS THE NUMBER OF MAX POINTS COLLECTED
IRSRU USES CHANNEL SPBUFF FOR DETECTIONS
SPMAX HOLDS THE DELTAS FROM SPBUFF
RDMAX HOLDS THE DELTAS FROM RCBUFF
ABMAX HOLDS THE DELTAS FROM ABUFF

```

CLEAR THE SCREEN
CALL TUCN
CALL TUCOFF

```

```

LOA BSEFLG
IF (.A,IS,ZERO)
LOA ISDFLG
IF (.A,IS,ZERO)
JMP ISOCALC
THIS WAS AN ISODLINE MANUEVER

```

```

LXI H,0
SHLD ROPT
ENDIF

```

```

FFDA 0
FSTA XSUM
FSTA YSUM
FSTA XSUM
FSTA XYSUM
FSTA XONT

```

```

LOOP1:
CALL GETX
CALL GETY
CALL XTPAIR

```

```

LHLD ROPT
LXI B,RSIZE
CALL PTRUPO
SHLD ROPT
DSKZ CNT,LOOP1

```

```

CALL LSTFIT
FLDA SLOPE
FABS
FSTA ABCAL
FFDA 1
FSTA BCAL

```

```

PRINT IT ON SCREEN
MVI A,4
STA OPCODE
CALL ANSOUT

```

```

CALL SAVECL
IF (PSU,IS,CARRY) THEN GOOD VALUES
MVI A,3
STA CALTYPE
MVI A,OFFH
STA LASTCAL (GOOD CALIBRATION)
XRA A
ORA
STA QUALFLG
CALL LPON
CALL PRTIME
PRM ISU,ISUPT
CALL LPOFF
ORA A
ENDIF

```

RET

QUALITATIVE ISOVOLUME CALIBRATION
 RCAL = 203
 ABCAL = 203

ISUPT: .WORD RCAL,ABCAL

GETX:

LXI H,ABBUFF
 LDOD ROPT
 DAD D
 MOV E,M
 INX H
 MOV D,M
 SDEO XVAL
 FALA XVAL
 FLOT
 FSTA XVAL
 RET

GETY:

LXI H,RCBUFF
 LDOD ROPT
 DAD D
 MOV E,M
 INX H
 MOV D,M
 SDEO YVAL
 FALA YVAL
 FLOT
 FSTA YVAL
 RET

XYPAIR:

FLOA XVAL
 FLOB XSUM
 FADD
 FSTA XSUM
 FLOB YVAL
 FLOA YSUM
 FADD
 FSTA YSUM
 FLOB XVAL
 FLOA XVAL
 FHL
 FLOB XSUM
 FADD
 FSTA XSUM

LDA CALTYPE
 IF (.A,IS,ZERO)
 MVI A,2
 STA OPCODE
 CALL ANSOUT
 RET
 ENOIF

HOW COLLECT DATA AS IF VALIDATING
 CALL VALCOL

LHLD MAXCNT
 LXI D,2
 ORA A
 ORSC D
 IF (PSW,IS,CARRY) THEN LESS THAN 2 BREATHS
 MVI A,1
 STA OPCODE
 CALL ANSOUT
 MVI A,0
 STA OPCODE (PRINT NOW ON SCREEN)
 CALL ANSOUT
 RET
 ENOIF

HOW FIND THE RATIOS OF SUM/NOON VOLUME
 CALL VALFRQ
 RETURNS WITH ADDRESS OF VALUES USED IN SUMMATION
 FSTA FOUT

HOW XVALUES HOLDS ALL OF THE RATIOS, XSUM HOLDS SUM OF ALL XVALUES
 XSUM HOLDS SUM OF SQUARES OF ALL XVALUES, AND FOUT HOLDS THE NUMBER
 OF VALUES USED IN THE SUMMATION

COMPUTE THE MEAN RATIO

FLOB XSUM
 FLOA FOUT
 FDIU
 FSTA MEAN

HOW KNOW THAT SUM NEEDS TO BE MULTIPLIED BY 1/MEAN, SO
 RCAL = CURRENT RC GAIN (RCGF) * 1/MEAN
 ABCAL = CURRENT AB GAIN (ABGF) * 1/MEAN

FLOA
 FADD
 FSTA
 FLOA MEAN
 FDIU
 FLOB ACGNF
 FHL
 FSTA RCAL
 FLOA
 FADD
 FSTA

FLOA XYSUM
 FADD
 FSTA XYSUM
 FLOA XONT
 FFOA
 FADD
 FSTA XONT
 RET

LSTFIT:

FLOA XSUM
 FLOB XSUM
 FHL
 FPSH
 FLOA XONT
 FLOB XSUM
 FHL
 FSWP
 FPOP
 FSUB
 FPSH
 FLOA XSUM
 FLOB YSUM
 FHL
 FPSH
 FLOA XONT
 FLOB XSUM
 FHL
 FSWP
 FPOP
 FDIU
 FSTA SLOPE
 RET

DOC2:

THIS CAN BE DONE AFTER EITHER A DOC1 (CALTYPE = 1/2)
 OR AFTER AN ISOVOLUME (CALTYPE = 3/4)
 IF CALTYPE = 0, THEN HAVEN'T CALIBRATED THE RC TO AB

FATB MEAN
 FLOA
 FDIU
 FHL
 FSTA ABCAL

MVI A,4
 STA OPCODE
 CALL ANSOUT

CALL SAVECL
 IF (PSW,IS,CARRY) THEN GOOD
 CALL LPON
 CALL PRTIME
 FPRN VL2,VL2PT
 CALL LPOFF
 LXI H,0
 SHLO MSERS
 XRA A
 STA QUALFLG (NOT QUALITATIVE, IT IS QUANTITATIVE CAL)

LDA CALTYPE
 IF (.A,LT,3) THEN DOC METHOD
 MVI A,2 (NOW ITS DOC QUANT)
 STA CALTYPE
 MVI A,0FFH
 STA LASTCAL (GOOD CAL
 ELSE
 MVI A,4
 STA CALTYPE (NOT ITS ISOV QUANT)
 MVI A,0FFH
 STA LASTCAL (GOOD CAL
 ENOIF

MULT THE CURRENT VALUE OF VTLIMIT BY 1/MEAN SO WILL BE SCALED ON
 NOTE: VTLIMIT IS INITIALLY SET WHEN DOING DOC1

THIS IS FOR INITIALIZING THE HYPONEA VOLUME FOR THE 1ST 5 MINUTES:

FLOB VTLIMIT
 FLOA MEAN
 FDIU
 FSTA VTLIMIT

PROBLEM: WHAT IF DID ISOVOLUME BEFORE HERE, THEN VTLIMIT WAS NOT
 EVER INITIALIZED ???
 ENOIF


```

    LXI    D,RCBUFF
    CALL   PRNG

    LHLD   ABAD ;ABDCHEN VALUE
    MOV    A,H
    CMA
    MVI    OFH
    MOV    M,A
    MOV    A,L
    CMA
    MOV    C,A
    SHLD   DATA
    SHLD   ABV

    LHLD   RIPT
    LXI    D,ABBUFF
    CALL   PRNG

; SUM INTO SPUFF--- TO PICK OFF BREATHS
    LHLD   RCV
    LOED   ABV
    DAD    D
    SHLD   DATA

    LHLD   RIPT
    LXI    D,SPUUFF
    CALL   PRNG

; INCREMENT RIPT FOR NEXT TIME
    LHLD   TIME
    INX    M
    SHLD   TIME
    SHLD   DATA
    LHLD   RIPT
    LXI    D,THRUFF
    CALL   PRNG

    LHLD   RIPT
    SHLD   ODRIPT

    LXI    B,RSIZE
    LHLD   RIPT
    CALL   PTRUPD
    SHLD   RIPT

    LHLD   ROPT
    CALL   FIND
    IF (PSW,IS,CARRY) ;NOT A MAX
        JMP    OUTER
    ENDDIF

; HAVE 1ST MAX
    LDA    FRSTMX
    CMA
    STA    FRSTMX
    SHLD   SPTENP ;TEMP CRITICAL POINT
    SOED   RCTENP
    SBOD   ABTENP
    LDA    PTF
    CMA
    STA    PTF ;NOW LOOK FOR MIN
    JMP    OUTER
    ENDDIF

    LHLD   ROPT
    CALL   FIND
    IF (PSW,IS,CARRY)
        SHLD   SPVALUE
        SOED   RCVALUE
        SBOD   ABVALUE
        LDA    PTF
        IF (.A,IS,ZERO) ;THIS IS A MAX POINT
            LHLD   RCVALUE
            SHLD   RCHDVAL
            LHLD   ABVALUE
            SHLD   ABHVAL
        CALL   DLTSTORE
        ; ONLY STORE THE MCA VARIABLES AT A MAX (ONLY WANT FOR INSP)
        CALL   MCASTORE
    ENDIF

; IF OLTIPTR = 1000, THEN EXIT
    LHLD   OLTIPTR
    LXI    D,1000
    CALL   TONE
    IF (PSW,IS,ZERO)
        MVI    A,OFFH
        STA    DLTFULL
        LXI    H,0
        SHLD   OLTIPTR
        JMP    CLKSTP
    ENDDIF
    ELSE
        ; THIS IS MIN POINT
        LHLD   RCVALUE
        SHLD   RCHVAL
    ENDIF

```

```

    IF (.A,IS,ZERO) ;SEE IF RIPT RESET TO 0
        LXI    D,0
        LHLD   RIPT
        CALL   TONE
        IF (PSW,IS,ZERO)
            MVI    A,OFFH
            STA    WRAPP
        ENDDIF
    ENDDIF

    LDA    BSEPLG
    IF (.A,IS,ZERO)
        LDA    ISOPLG
        IF (.A,IS,ZERO) ;THEN NO PEAK DETECTION
            JMP    EXIT
        ENDDIF
    ENDDIF

    LDA    SLOPEF
    IF (.A,IS,ZERO) ;THEN INIT POINTS NOT COLLECTED YET
        CALL   GETP2 ;RETURNS C IF < 3 PTS IN BUFFER
        ; THIS JUST MAKES SURE EXTRA
        ; POINTS IN BUFFER
        JC    EXIT ;NOT ENOUGH POINTS IN BUFFER

; NOTE: GETP2 INITIALIZES ROPT TO LINE UP WITH DERIV STUFF
; SKIP 1ST 3 POINTS IN BUFFER
    LXI    H,4 ;OFFSET POINTING TO POINT 4
    SHLD   ROPT

    MVI    A,OFFH
    STA    SLOPEF
    ENDDIF

    CALL   DERIVS

;.....
; MUST ADD THE CODE TO DO PEAK DETECTION
; HERE TOO !!

    LDA    FRSTMX
    IF (.A,IS,ZERO) ;HAVE NOT FOUND 1ST MAX POINT

    LHLD   ABVALUE
    SHLD   ABHVAL
    LDA    FRSTMX
    IF (.A,IS,ZERO) ;THIS IS THE 1ST MIN, DO NOT STORE
        LDA    FRSTMX
        CMA
        STA    FRSTMX
        ELSE
            ;STORE THE DELTA
            CALL   DLTSTORE
        ENDIF

; IF OLTIPTR = 1000, THEN EXIT
    IF OLTIPTR = 1000, THEN EXIT
        LHLD   OLTIPTR
        LXI    D,1000
        CALL   TONE
        IF (PSW,IS,ZERO)
            MVI    A,OFFH
            STA    DLTFULL
            LXI    H,0
            SHLD   OLTIPTR
            JMP    CLKSTP
        ENDDIF
    ENDDIF

    ENDDIF
    LDA    PTF
    CMA
    STA    PTF
    ENDDIF

;.....
; UPDATE ROPT FOR NEXT TIME
    LHLD   ROPT
    LXI    B,RSIZE
    CALL   PTRUPD
    SHLD   ROPT

; EXIT:
; LOOK FOR ENTER
    CALL   CSTS
    IF (PSW,IS,MZERO)
        CALL   GINGP
        IF (.A,EQ,00H) ;IF ENTER
            JMP    CLKSTP
        ENDDIF
    ENDDIF

    XRA    A

```



```

ELSE
  CALL WTSWECAL
ENDIF
IF (PSM,IS,CARRY) THEN BAD CALS
  MVI A,0
  STA CALTYPE ;CALS TO 1 AND 1
  MVI A,0
  STA LASTCAL ;NO GOOD CAL
  STA GOODBASE ;NO GOOD BASELINE NOW
;PRINT OUT WHAT THE BAD CALS WERE AND THE STDS
  CALL BOCALS
  STC
  RET
ENDIF

;NOW MUST USE THE MEAN VT MEASURED TO SCALE THE CAL FACTORS SO
; MEAN VT = VT00/10
  CALL RESCALE

  LDA VTFL0
  IF (A,IS,ZERO)
    MVI A,1
    STA DPCODE
    CALL HISCOUT ;DISPLAY THE CALS ON THE MONITOR
  ENDIF

  LDA VTFL0
  IF (A,IS,ZERO) ;WANT TO PRINT ERROR CODE IF BAD
    CALL SWECL
  ELSE
    CALL WTSWECAL
  ENDIF
  IF (PSM,IS,NCARRY) THEN GOOD CALS
    CALL LPON
    CALL PRTIME
    FPRN V1,V2PT
    CALL LPOFF
    CALL CHMCA ;COMPUTE MCA/VT
    CALL LPON
    FPRN V2,V2PT
    CALL PRTHYP
    CALL LPOFF
    MVI A,1
    STA CALTYPE ;GOOD QUAL
    MVI A,0FFH
    STA LASTCAL
    ORA A
    ;GOOD CALS
  ELSE
    MVI A,0
    ;NO GOOD CALIBRATION (WHEN BEGAN THIS RUN, RESET
    STA CALTYPE ;CALS TO 1 AND 1

```

```

  CALL CHPSTATS
  IF (PSM,IS,ZERO) NOT ENOUGH DATA
    MVI A
    STA GOODBASE ;NOT A GOOD BASELINE
    STC
    RET
  ENDIF

  CALL RESCALE ;THIS WILL CHANGE VALUE OF RECAL AND ABCAL

  LDA CALTYPE
  IF (A,IS,0) ;IS0/ QUAL

;SAVE CALS
;NOW MUST USE THE MEAN VT MEASURED TO SCALE THE CAL FACTORS SO
; MEAN VT = VT00/10
  CALL WTSWECAL

  IF (PSM,IS,NCARRY) THEN GOOD CALS
    MVI A,0
    STA CALTYPE ;IS0-QUAL WITH BASELINE
    MVI A,0FFH
    STA LASTCAL ;GOOD CAL
    CALL LPON
    CALL PRTIME
    FPRN 831.851PT
    CALL LPOFF
    CALL CHMCA ;COMPUTE MCA/VT
    CALL LPON
    FPRN V2,V2PT
    CALL PRTHYP
    CALL LPOFF
    MVI A,0FFH
    STA GOODBASE ;GOOD BASELINE
    ORA A
    ;GOOD CALS
  ELSE
    FFOA I
    FSTA RECAL
    FSTA ABCAL
    CALL CHMCA ;COMPUTE MCA/VT
    CALL LPON
    LXI H,STARS
    CALL TXTP
    FPRN 831.851PT
    FPRN V2,V2PT
    CALL PRTHYP
    LXI H,STARS
    CALL TXTP
    CALL LPOFF
    MVI A,0
    STA GOODBASE ;BAD BASELINE

```

```

;PRINT OUT WHAT THE BAD CALS WERE AND THE STDS
  CALL BOCALS
  FFOA I
  FSTA RECAL
  FSTA ABCAL
  CALL CHMCA ;COMPUTE MCA/VT
  CALL LPON
  FPRN V2,V2PT
  CALL PRTHYP
  LXI H,STARS
  CALL TXTP
  CALL LPOFF

  STC
  ENDIF
  RET

  BOCALS:
  CALL LPON
  LXI H,STARS
  CALL TXTP
  CALL PRTIME
  FPRN BADC,BADPT
  CALL LPOFF
  RET

  PRTHYP:
  FPRN HYP,HYPPT
  LDA QUALFLO
  IF (A,IS,ZERO) ;QUAL
    LXI H,RYPT
  ELSE ;QUANT
    LXI H,ANYPT
  ENDIF
  CALL TXTP
  RET

```

```

;BOCMP:
;THIS COMPUTE MEAN VT AND IF IS0W WAS QUALITATIVE.. CALTYPE = 3,
; THEN IT SCALES IT TO VT00/10

```

```

  STC
  ENDIF
  ELSE
    ;DO NOT DID QUANT-1500
    IF (A,IS,0)
      MVI A
      STA CALTYPE ;IS0 QUANT WITH BASELINE
      CALL LPON
      CALL PRTIME
      FPRN 831.851PT
      CALL LPOFF
    ;CALS DO NOT CHANGE
    FFOA I
    FSTA RECAL
    FSTA ABCAL
    CALL CHMCA ;COMPUTE MCA/VT
    CALL LPON
    FPRN V2,V2PT
    CALL PRTHYP
    CALL LPOFF
    MVI A,0FFH
    STA GOODBASE ;GOOD BASELINE
    ORA A
    ;GOOD CALS
  ELSE
    IF (A,IS,0) ;TYPED IN CALS. QUANT. JUST SETTING BASELINE STUFF
      LDA QUALFLO
      IF (A,IS,ZERO)
        CALL LPON
        CALL PRTIME
        FPRN 831.851PT
        CALL LPOFF
      ;CALS DO NOT CHANGE
      FFOA I
      FSTA RECAL
      FSTA ABCAL
      CALL CHMCA ;COMPUTE MCA/VT
      CALL LPON
      FPRN V2,V2PT
      CALL PRTHYP
      CALL LPOFF
      MVI A,0FFH
      STA GOODBASE ;GOOD BASELINE
      ORA A
      ;GOOD CALS
    ELSE
      ;TYPED IN CALS. QUAL SO WILL CHANGE CALS
    ;
    ;SAVE CALS
    ;NOW MUST USE THE MEAN VT MEASURED TO SCALE THE CAL FACTORS SO
    ; MEAN VT = VT00/10
    CALL WTSWECAL
  ;
  IF (PSM,IS,NCARRY) THEN GOOD CALS
    CALL LPON

```

```

CALL LPOFF
CALL CHPMCA :COMPUTE MCA/VT
CALL LPOH
PPRN V2,V2PT
CALL PRTHYP
CALL LPOFF
MVI A,BFFH
STA GOODBASE 16000 BASELINE
ORA A 16000 CALS
ELSE
FFQA 1
FSTA RECAL
FSTA ASCAL
CALL CHPMCA :COMPUTE MCA/VT
CALL LPOH
CALL PRTHYP
PPRN SSBA0,SSBPT
PPRN V2,V2PT
CALL PRTHYP
CALL LPOFF
MVI A,B
STA GOODBASE 1800 BASELINE
STC 1800 CALS
ENDIF

```

CHPMCA:

```

:COMPUTE MEAN MCA/VT
:MUST REMEMBER THAT ALL RC VALUES MUST BE MULT BY RECAL
:AND ALL AB VALUES MUST BE MULT BY ASCAL
NOTE: WILL NOT BE ABLE TO COMPUTE MCA/VT THE CLEAN WAY OF USING
THEN SUM OF THE SUM'S DERIV BECAUSE THE CAL FACTOR PROBLEM
SO WILL HAVE TO DIVIDE BY RC DELTA * RECAL + AB DELTA * ASCAL
LXI H,0X0000
SHLD ADDR1
LDA DLTFULL
IF C.A.,IS,ZERO) :FULL
SHLD DLTIPTR
SHLD OFFSET

```

```

PHUL MCAVALUE
:FSTA MCAVALUE
:GET VT VALUE
LHLD OFFSET
LXI D,RCDATA
DAD D
MOV E,H
INX H
MOV D,H
SDED FRC
LHLD OFFSET
LXI D,ABDATA
DAD D
MOV E,H
INX H
MOV D,H
SDED FAB
FILA FRC
FLOT RECAL
FLOB RECAL
PHUL FAB
FPSH FAB
FILA ASCAL
FLOT ASCAL
FLOB ASCAL
PHUL FAB
FATS
FPOP
FADD CCS
PHUL UTVALUE
FSTA UTVALUE

```

MCA SUM = SUM OF MCAVALUE/UTVALUE

```

FLOB MCAVALUE
FLOA UTVALUE
FOIU
IF 1, THEN MAKE = 1
FPSH
FATS
FFQA 1
FSUB
FTST ARSQ,LT
IF (PSM,IS,ZERO)
FPOP
FFQA 1
ELSE
FPOP

```

```

ENDIF
FFQA 0
FSTA UTSUM
FSTA MCASUM
FSTA XONT
LOOP
LHLD ADDR1
LOED MCA+ADDR
CALL TONE
IF (PSM,IS,ZERO)
RET
ENDIF
:GET MCA VALUE
LHLD ADDR1
MOV E,H
INX H
MOV D,H
SDED RECAL
INX H
MOV E,H
INX H
MOV D,H
SDED ABMCA
INX H
MOV E,H
INX H
MOV D,H
SDED SHMCA
INX H
SHLD ADDR1
FILA RECAL
FLOT RECAL
FLOB RECAL
PHUL ABMCA
FPSH ABMCA
FILA ASCAL
FLOT ASCAL
FLOB ASCAL
PHUL FATS
FPOP
FADD
FATS
FFQA 15 :BECAUSE OF 9 PT DERIV 40
FOIU :1/4 BECAUSE OF DIVISION WHEN COLLECTED
FLOB CCS

```

```

ENDIF
FLOB MCASUM
FADD
FSTA MCASUM
FLOB UTVALUE
FLOA UTSUM
FADD
FSTA UTSUM
FLOB XONT
FFQA 1
FADD
FSTA XONT
LHLD OFFSET
INX H
INX H
SHLD OFFSET
LOED DLTIPTR
CALL TONE
IF (PSM,IS,ZERO)
RET
ENDIF
:HOW OFFSET POINTS TO EXP DELTA---SKIP IT !!
LHLD OFFSET
INX H
INX H
SHLD OFFSET
LOED DLTIPTR
CALL TONE
IF (PSM,IS,ZERO)
RET
ENDIF
ENDLOOP
FLOB UTSUM
FLOA XONT
FOIU
FSTA UTVALUE :MEAN UT
FLOB MCASUM
FLOA XONT
FOIU
FSTA MCAVT :MEAN MCA/VT
MEAN MCAVT = MEAN MCA / MEAN UT

```



```

UTSUM1 ,BLKB 4
ACASUM1 ,BLKB 4
PK(AVG)1 ,BLKB 4
PK(XVG)1 ,BLKB 4
TISUM1 ,BLKB 4
PELOC
END

```

```

INSERT PPMAC.SRC
INSERT CALCOM.ASH
INSERT CXCOM
INSERT SELCOM
INSERT BAGCOM
INSERT VARCOM.CCC
INSERT SYCOM
INSERT HROCOM

EXTERN APBP,ZALL,AUDB1

EXTERN ATON,RTOP,ATICK
EXTERN TDHE
EXTERN DERIVS
EXTERN PTRUPO,GETP2,FINO,ROUND

EXTERN SPQ1,SPQF

ENTRY SETUP
CR = 00H
FINIT

```

```

SETUP:
;SETUP THE CLOCK FOR VALIDATION
;SAMPLE AT 20 PTS PER SEC
PUSH H
;INIT REAL-TIME BUFFER POINTERS
;MODIFIED 09-24-81
MVI R,3
STA WAITFLO

XRA A
STA UNAPP
STA SECFLD
CMA
STA BFLD
XRA A
STA VFRSTHOK
LXI H,0
SHLD ELPTH
CHLD ELPHIN
SHLD SECONDS
SHLD CPPT2

```

```

LXI H,0
SHLD RIPT
SHLD CORIPT
SHLD SPTEMP
SHLD RMSUM

LXI H,8 ;SKIP THE 1ST 4 DATA POINTS
SHLD ROPT

LXI H,THURFF
MVI H,0
JOK H
MVI H,0
LXI H,0
SHLD TIME
;INIT VARIABLES FOR PICKING MAX AND MINS
XRA A
STA PTF ;START BY LOOKING FOR MAX PT
STA SLOPEF ;INIT MAX NOT FOUND

XRA A
STA INTFLG
CMA
STA GFLG ;INDICATES CLOCK ON
CALL AUDEN
CALL APBP
CALL ZALL

LXI H,IRSRV
MVI A,20 ;CLOCK RATE
CALL RTOM ;TURN ON CLOCK

LDA BAG
IF (.A,IS,ZERO) ;USING SPIROMETER
CALL SPON
ENDIF

POP H
RET

```

```

CLXSTP:
;STOP DATA COLLECTION
;TURN CLOCK OFF
;MODIFIED 09-24-81
CALL RTOP ;TURN OFF CLOCK
LDA BAG

```

```

IF (.A,IS,ZERO) ;USING SPIROMETER
CALL SPBP
ENDIF
XRA A
STA GFLG
CALL AUDEN
CALL APBP

;*****
POP P24
POP H
POP D
POP B
RET

```

```

IRSRV:
;INTERUPT SERVICE ROUTINE

PUSH B
PUSH D
PUSH H
PUSH PSW

CALL RTICK ;SERVICE INTERRUPT
LDA INTFLG
IF (.A,IS,ZERO) ;THEN INTERRUPT INTERRUPT
NOP
ELSE
CMA
STA INTFLG ;SET FLAG
ENDIF

CALL CLKTIK

CALL SAMPLE

LDA SLOPEF
IF (.A,IS,ZERO)
CALL GETP2 ;RETURNS C IF < 5 PTS IN BUFFER
;THIS JUST MAKES SURE EXTRA
;POINTS IN BUFFER
JC EXIT ;NOT ENOUGH POINTS IN BUFFER

```



```

CALL DERIVS
LDA UFRSTOX
IF (A,IS,ZERO) ;HAVE NOT FOUND 1ST MAX
LHLD ROPT
CALL FIND
IF (PSM,IS,CARRY) ;NOT MAX
LHLD ROPT
LXI B,RSIZE
CALL PTRUPD
SHLD ROPT
JMP EXIT
ENDIF
;MAX FOUND
LDA UFRSTOX
ORA STA
LDA UFRSTOX
ORA PTF
ORA STA
STA PTF
SHLD SPTMP ;TEMP CRIT PT
SOED RCTEMP ;CORRESPONDING RC PT
SBCD ARTEMP ;CORRESPONDING AB PT
LHLD ROPT
LXI B,RSIZE
CALL PTRUPD
SHLD ROPT
JMP EXIT
ENDIF

```

```

;INIT MAX FOUND
LHLD ROPT
CALL FIND

```

```

;UPDATE ROPT FOR NEXT TIME

```

```

PUSH PSM
PUSH O
PUSH S
PUSH H
LHLD ROPT
LXI B,RSIZE
CALL PTRUPD
SHLD ROPT
POP H
POP S

```

```

LHLD SPVALUE ;SUM VALUE
JMP L1001
ENDIF
L1001: LHLD ABVALUE
MOV M,E
INX M
MOV M,D
LHLD MAXCNT
INX H
SHLD MAXCNT
LHLD CPPTR
INX H
SHLD CPPTR
LHLD MAXCNT
LXI D,BATHX
ORA A
SBCD D
IF (PSM,IS,ZERO) ;THEN BUFFER FULL
JMP CLKSTP
ENDIF
ELSE
LHLD CPPTR
LXI D,SPMIN
DAD D
LDA BAG
IF (A,IS,ZERO) ;THEN USING SPIROMETER, AND
;SPVALUE HOLDS SPIROMETER VALUE
LHLD SPVALUE
ELSE ;USING SPIROBAG, SO MIN = 0
LXI D,0
ENDIF
MOV M,E
INX M
MOV M,D
LHLD CPPTR
LXI D,RCMIN
DAD D
LHLD RQVALUE
MOV M,E
INX M
MOV M,D
LHLD CPPTR
LXI D,ABMIN
DAD D

```

```

;FIND RETURNED C SET IF CP FOUND
;AND HL HOLDS SP PT VALUE
;AND DE HOLDS RC PT VALUE
;AND BC HOLDS AB PT VALUE

```

```

IF (PSM,IS,CARRY)
SHLD SPVALUE
SOED RQVALUE
SBCD ABVALUE

```

```

LDA PTF
IF (A,IS,ZERO) ;THEN MAX FOUND

```

```

ORA A
ORA STA
STA BFLD
LXI D,SPMAX
LHLD CPPTR
DAD D

```

```

LDA BAG
IF (A,IS,ZERO) ;THEN USING SPIROMETER, AND
;SPVALUE HOLDS SPIROMETER VALUE
LHLD SPVALUE
ELSE ;USING SPIROBAG, SO FIXCU HOLDS VALUE
LHLD FIXCU
ENDIF

```

```

MOV M,E
INX M
MOV M,D

```

```

LHLD CPCTR
LXI D,RCMAX
DAD D
LHLD RQVALUE
MOV M,E
INX M
MOV M,D

```

```

LHLD CPCTR
LXI D,ABMAX
DAD D

```

```

;IF VALIDATING AND IF USING SPIROBAG
;THEN SPVALUE HOLDS THE SUM MAX
LDA BAG
IF (A,IS,ZERO) ;THEN USING SPIROBAG

```

```

LHLD DATA
MOV A,H
ORA A
RRI OFH
MOV M,A
MOV A,L
ORA A
MOV L,A
SHLD DATA

```

```

ENDIF
RET

```

```

;PRND
;HL HOLDS THE RING INPUT PTR
;DE HOLDS ADDR OF RING

```

```

DAD D
LHLD DATA
MOV M,E
INX M
MOV M,D
RET
;END

```


.END

```

INSERT CALCON
INSERT SELCON
INSERT EFFCON
INSERT DRACON
INSERT SIOCON
INSERT FPMAC.SRC
INSERT PKCON.ASM
ENTRY FIND,GETP3,DERIVS
EXTEND TONE,PTAUPO
FINIT

```

FLIND:

```

; HL = OFFSET OF CURRENT PT ON ENTRANCE
; ON EXIT: IF CP FOUND, THEN CARRY SET AND
; HL = SPIROMETER PT
; DE = R18 CAGE PT.
; BC = ADDRESS PT
SHLD CUROFF

;
LHLD CUROFF
LXI D,SPBUFF
DAD D
MOV E,M
INX H
MOV D,M
DDED CURSP

;
LHLD CUROFF
LXI D,RCBUFF
DAD D
MOV E,M
INX H
MOV D,M
DDED CURRC

;
LHLD CUROFF
LXI D,ABSBUFF
DAD D
MOV E,M
INX H
MOV D,M
DDED CURAB

;
LHLD CUROFF
LXI D,TNIBUFF
DAD D
MOV E,M
INX H
MOV D,M

```

```

DDED CURTH

CALL GTSIGN ;CHECK SIGNS OF DERIVATIVES

```

```

LDA PTF
IF (A,13,1ZER0) ;LOOK FOR MIN
LHLD SPTEMP
DDED CURSP
ORA A
DSEC 0
IF (PSW,13,1CARRY) ;THEN TRIP MIN FOUND
LHLD CURSP
SHLD SPTEMP
LHLD CURRC
SHLD RCTEMP
LHLD CURAB
SHLD ABTEMP
LHLD CUROFF
SHLD SPTTEMP
LHLD CURTH
SHLD THTEMP

```

```

11. ADD CURRENT RCDERV PT TO RC COUNTS
12. ADD RCTEMP TO RC COUNTS
13. RCTEMP == 0
14. ADD CURRENT ABDERV TO AB COUNTS
15. ADD ABTEMP TO AB COUNTS
16. ABTEMP == 0

```

CALL UPDT1

```

ORA A
RET
ELSE
;IMPOSSIBLE CP SO CHECK SLOPE BETWEEN
;CURRENT PT AND TEMP PT
LHLD CURSP
DDED SPTEMP
ORA A
DSEC 0
;NOW HL HOLDS DIFFERENCE
DDED MINCE
ORA A
DSEC 0
IF (PSW,13,1CARRY) ;THEN SPTEMP IS CP
LHLD OFFTEMP
SHLD CUROFF
LHLD SPTEMP
PUSH H
LHLD RCTEMP
PUSH H
LHLD ABTEMP
PUSH H

```

```

LHLD CURSP
SHLD SPTEMP
SHLD CURRC
SHLD RCTEMP
LHLD CURAB
SHLD ABTEMP
LHLD CUROFF
SHLD OFFTEMP
LHLD THTEMP
SHLD CPTIME ;TIME OF THIS CP

LHLD CURTH
SHLD THTEMP ;NEJ TEMP

```

```

11. STORE RC COUNTS
12. RC COUNTS == RCTEMP
13. RCTEMP == 0
14. STORE AB COUNTS
15. AB COUNTS == ABTEMP
16. ABTEMP == 0

```

CALL UPDT3

```

POP B ;AB VALUE
POP D ;RC VALUE
POP H ;SP VALUE
STC ;SET CARRY
RET

```

ELSE
;NOT A LARGE ENOUGH SLOPE

11. ADD RCDERV TO RCTEMP

12. ADD ABDERV TO ABTEMP

CALL UPDT3

ORA A

RET

ENDIF

ELSE ;LOOK FOR MAX

LHLD SPTEMP

DDED CURSP

XCHG

ORA A

DSEC 0

IF (PSW,13,1CARRY) ;THEN CURRENT = SPTEMP

```

LHLD CURSP
SHLD SPTEMP
LHLD CURRC
SHLD RCTEMP
LHLD CURAB
SHLD ABTEMP
LHLD CUROFF
SHLD OFFTEMP

```



```

ENDIF
      RET
UPDT3:
1. STORE RC COUNTS
2. RC COUNTS (== RCTCNT)
3. RCTCNT (== 0)
4. STORE AB COUNTS
5. AB COUNTS (== ABTCNT)
6. ABTCNT (== 0)
   LHL RCIN
   SHL RCINMOL
   LHL RCTCNT
   SHL RCIN
   LXI H,0
   SHL RCTCNT
   LHL ABIN
   SHL ABINMOL
   LHL ABTCNT
   SHL ABIN
   LXI H,0
   SHL ABTCNT
SUM: DO NOT SEPERATE INPHASE AND OUT OF PHASE
   LHL SH10
   SHL SH10MOL
   LHL SHTCNT
   SHL SH10
   LXI H,0
   SHL SHTCNT
FOR PEAK FLOW STUFF:
SHDERV = CURRENT F.L.      SHD = ABS(SHDERV)

IF (A,15,ZER0) FOR INSP SIDE
TEMPK IS THE PEAK FLOW, AND PYTIME IS THE TIME OF IT
STORE THE TIME OF THE PEAK FLOW
CALL PKSTORE
LHL TTNEG
SHL TTNEG
LHL TTTPX
SHL TTTPX
LHL TTTPH
SHL TTTPH
LXI H,0
SHL TTPOS
SHL TTNEG
LHL CURTH
SHL CURTH
LHL TTTHH
SHL TTTHH
ELSE
FOR E.P. SIDE
TEMPK IS THE PEAK FLOW, AND PYTIME IS THE TIME OF IT
STORE THE TIME OF THE PEAK FLOW
CALL PKSTORE
LHL TTPOS
SHL TTPOS
LHL TTTPX
SHL TTTPX
LHL TTTPH
SHL TTTPH
LXI H,0
SHL TTPOS
SHL TTNEG
LHL CURTH
SHL CURTH
LHL TTTHH
SHL TTTHH
ENDIF
      RET
UPDT3:
1. ADD ABS(RCDERV) TO RCTCNT
2. ADD ABS(ABDERV) TO ABTCNT
   LHL ABSRCO
   LOED RCTCNT
   DAD 0
   SHL RCTCNT
   LHL ABSABO
   LOED ABTCNT
   DAD 0
   SHL ABTCNT
SUM: DO NOT SEPERATE INPHASE AND OUT OF PHASE
   LHL ABS10
   LOED SHTCNT
   DAD 0
   SHL SHTCNT
FOR PEAK FLOW STUFF:
SHDERV = CURRENT FLOW, ABS10 = ABS(SHDERV)
IF CURRENT FLOW > 0
   LHL SHDERV
   LOED ABS10
   CALL TONE
   IF (PSW,15,CARRY)
   IF CURRENT FLOW > 0
      LHL TTPOS
      LOED SHDERV
      ORA A
      DSHC 0
      IF (PSW,15,CARRY)
      LHL SHDERV
      SHL TTPOS
      LHL CURTH
      SHL TTTHH
      ENDIF
   ELSE
   IF ABS(CURRENT FLOW) > TTNEG
      LHL TTNEG
      LOED ABS10
      ORA A
      DSHC 0
      IF (PSW,15,CARRY)
      LHL ABS10
      SHL TTNEG
      LHL CURTH
      SHL TTTHH
      ENDIF
   ENDIF
      RET
PKSTORE:
LHL PKTIME
SHL PKTIME
RET
DERIV:
*****
DERIV USING 9 POINTS

```

```

;SRCOFST HOLDS OFFSET TO GET Y'S FROM RAM DATA BUFFERS
;ASSUME: RCY1,RCY2,RCY3,RCY4, ABY1,ABY2,ABY3,ABY4 ALREADY SET
;
LXI 0,RCBUFF
LHLD SRCOFST
DAD D
MOV E,H
IIOX H
MOV D,H
SHLD RCY9
;
LHLD SRCOFST
LXI 0,ABBUFF
DAD D
MOV E,H
IIOX H
MOV D,H
SHLD ABY9
;
LXI 9,RSIZE UPDATE FOR NEXT TIME
LHLD SRCOFST
CALL PTRUPD
SHLD SRCOFST
;
;DERIV = (7-1)*4 + (3-2)*1 + (7-3)*2 + (4-4)
LHLD RCY9
LDED RCY1
ORA A
DSBC D
DAD H ;(RCY9-RCY1)+2
DAD H ;(RCY9-RCY1)+4
PUSH H
;
LHLD RCY8
LDED RCY2
ORA A
DSBC D
MOV E,L
MOV D,H
DAD H ;+2
DAD D ;+3
POP D
DAD D
PUSH H
;
LHLD RCY7
LDED RCY3
ORA A

```

```

LHLD ABY4
LDED ABY4
ORA A
DSBC D
POP D
DAD D
SHLD ABDERV
;
;***** DIVIDE RCDEPV BY 4 TO AVOID OVERFLOW WHEN INTEGRATING IT
LHLD ABDERV
IPAR H
IPAR L
IPAR H
IPAR L
SHLD ABDERV
;
;***
;
;RESET VALUES FOR NEXT TIME
LHLD RCY2
SHLD PCY1
LHLD RCY3
SHLD RCY2
LHLD RCY4
SHLD PCY3
LHLD RCY5
SHLD RCY4
LHLD RCY6
SHLD RCY5
LHLD RCY7
SHLD RCY6
LHLD RCY8
SHLD PCY7
LHLD RCY9
SHLD RCY8
;
LHLD ABY2
SHLD ABY1
LHLD ABY3
SHLD ABY2
LHLD ABY4
SHLD ABY3
LHLD ABY5
SHLD ABY4
LHLD ABY6
SHLD ABY5
LHLD ABY7
SHLD ABY6
LHLD ABY8
SHLD ABY7
SHLD ABY8

```

```

DAD D
PUSH H
;
LHLD RCY4
LDED RCY4
ORA A
DSBC D
POP D
DAD D
SHLD RCDEPV
;
;***** DIVIDE RCDEPV BY 4 TO AVOID OVERFLOW WHEN INTEGRATING IT
LHLD RCDEPV
IPAR H
IPAR L
IPAR H
IPAR L
SHLD RCDEPV
;
;***
;
;DERIV = (7-1)*4 + (3-2)*1 + (7-3)*2 + (4-4)
LHLD ABY1
LDED ABY1
ORA A
DSBC D
DAD H ;(ABY9-ABY1)+2
DAD H ;(ABY9-ABY1)+4
PUSH H
;
LHLD ABY8
LDED ABY2
ORA A
DSBC D
MOV E,L
MOV D,H
DAD H ;+2
DAD D ;+3
POP D
DAD D
PUSH H
;
LHLD ABY7
LDED ABY3
ORA A
DSBC D
DAD H
POP D
DAD D
PUSH H

```

```

LHLD RCDEPV
LDED ABDEPV
DAD D
SHLD SMOERV
;
;
;COMPUTE ABSOLUTE VALUE OF RCDEPV AND ABDEPV AND SMOERV
LHLD RCDEPV
MOV A,H
ANI 80H
IF (A,IS,ZERO) ;THEN RCDEPV < 0
MOV A,L
ORA A
MOV L,A
MOV A,H
ORA A
MOV H,A
INX H
SHLD ABSRCO
ELSE
SHLD ABSACO
ENDIF
;
LHLD ABDEPV
MOV A,H
ANI 80H
IF (A,IS,ZERO) ;THEN ABDEPV < 0
MOV A,L
ORA A
MOV L,A
MOV A,H
ORA A
MOV H,A
INX H
SHLD ABSABO
ELSE
SHLD ABSABO
ENDIF
;
LHLD SMOERV
MOV A,H
ANI 80H
IF (A,IS,ZERO) ;THEN SMOERV < 0
MOV A,L
ORA A
MOV L,A
MOV A,H
ORA A
MOV H,A
INX H
SHLD ABSSHO

```

```

LHLD RIPT
LXI 0,20
ORA A
DSBC 0
IF (PSW.IS.CARRY) THEN ( 10 PTS
    STC
ELSE

```

```

      LXI      H,PC,BUFF
      MOV     E,M
      INC     H
      MOV     D,M
      SDEC
      RCTI

```

```

1      IROK      H
      MOV      E,M
      IROK      H
      MOV      D,M
      SOED      RCY2

```

```

      LIX      H
      MOV      E,M
      LIX      H
      MOV      O,M
      SDED     RCY3

```

170X	H
MOJ	E.M
110X	H
MOJ	C.M
SDER	RCY4

MOU	D,M
SOED	ABY3
1	
1IX	H
MOU	E,M
1IX	H
MOU	D,M
SOED	ABY4

1

170X	H
MOU	E.M
170X	H
MOU	D.M
50ED	4BY7

TICK	H
MOV	E,M
POP	H
MOV	D,M
SPED	ARY

```

LAI      H.14      :OFFSET FOR THE 9TH VALUE IN BUFFERS
SHLD     SRCOFST

```

LX1	H.8	1 OFFSET OF MIDDLE OF CERT.
SHLD	RCPT	

LXI N.O
SHLO INTERC
GRA A

85

.EXTERN CFO
.LGC CFO

```

CUROFF:      .BLKB      2
OFFTEMP:     .BLKB      2
               .RELOC

```

.....

```
.IDENT :OKFIND
.INSERT SP00.ASM
.INSERT CALCOM
.INSERT FPMAC.GRC
.INSERT EFFCOM
.INSERT DXCOM
.INSERT DRVCOM
.INSERT SIOCOM
```

ITW	E.O.
IPX	H
MOU	D.M
SDPO	RCY9

IPX	M
MOV	S.M
IPX	M
MOV	O.M
SDO	RCY4

POX	4
MOV	E.M
POX	H
MOV	D.M
SDSD	RCY7

INX	M
MOV	E.M
IPX	M
MOV	D.M
SDPD	RCYS

```

LXI      H,ABBUFF
MOV      E,H
POX      H
MOV      D,H
JDEQ     ABY1

```

IPX	H
MOV	Z.M
IPX	H
MOV	D.M
SPED	ABY2

IPX	H
MOV	E.M
IPX	H
MOV	D.M
SDOD	ABYJ

FOX	M
NOV	E.M
INC	M
77	O.M
DEED	ABY

IND	H
MOU	E.M
IPX	H

```

.ENTRY  OXFIND.DKGETP2
.EXTEND  TONE
.ENTRY  DERIVS
.EXTEND  PTRAPD
FINIT

```

1 ASSUMED THAT ALL RC VALUES, A3 VALUES, AND SUM VALUES
1 WILL HAVE FIRST 4 BITS = 33000 COMING FROM
1 A-0.

ONF LID

IML = OFFSET OF CURRENT PT ON ENTRANCE

1
1 G4 EXIT: IF CP FOUND, THEN CHARY SET #40

HL = SUN PT
DE = BIR CAGE PT

DE = RIB CAGE PT
BC = AEROMEN 3rd

1
1

```

PUSH    M
PUSH    M
PUSH    M
LXI     D,SPBUFF
DAD      D
MOV      E,M
INX      H
MOV      O,M
SPED     CURSP

```

POP	H
LXI	O,RCBUFF
DAO	D
MOV	E,M
INX	H
MOV	D,M
SDED	CURRC

```
POP      H
LXI      D,ASSUFF
DAD      D
MOV      E,M
INX      H
MOV      D,M
JNE      CURAR
```

```
POP      H
LXI      D,THUBFF
DAD      D
MOV      E,H
INX      H
MOV      D,H
```



```

CALL GTSIGN CHECK SIGNS OF DERIVATIVES

LDA PTF
IF (.A,IS,ZERO) LOOK FOR MIN
LHLD SPTEMP
LOED CURSP
ORA A
DSCC 0
IF (PSM,IS,NCARRY) THEN TRIP MIN FOUND
LHLD CURSP
SHLD SPTEMP
LHLD CURRC
SHLD RCTEMP
LHLD CURAB
SHLD ABTEMP
LHLD CURTH
SHLD SHTIME

11. ADD CURRENT RC DERIV PT TO RC COUNTS
12. ADD RCTOINT TO RC COUNTS
13. RCTOINT <= 0
14. ADD CURRENT AB DERIV TO AB COUNTS
15. ADD ABTOINT TO AB COUNTS
16. ABTOINT <= 0
CALL UPDT1

ORA A
RET
ELSE
;POSSIBLE CP SO CHECK SLOPE BETWEEN
;CURRENT PT AND TEMP PT
LHLD CURSP
LOED SPTEMP
ORA A
DSCC 0
;NOW HL HOLDS DIFFERENCE
LOED MINCC
ORA A
DSCC 0
IF (PSM,IS,NCARRY) THEN SHTEMP IS CP
LHLD SPTEMP
PUSH 5
LHLD RCTEMP

IF (PSM,IS,NCARRY) THEN CURRENT IS SHTEMP
LHLD CURSP
SHLD SPTEMP
LHLD CURRC
SHLD RCTEMP
LHLD CURAB
SHLD ABTEMP
LHLD CURTH
SHLD SHTIME
CALL UPDT1

CPA A
RET
ELSE ;POSSIBLE CP
LHLD SPTEMP
LOED CURSP
ORA A
DSCC 0
;NOW HL = DIFF
LOED MINCC
ORA A
DSCC 0
IF (PSM,IS,NCARRY) THEN SHTEMP IS CP
LHLD SPTEMP
PUSH 5
LHLD RCTEMP
PUSH 5
LHLD ABTEMP
PUSH 5
LHLD SHTIME
SHLD TVALUE
LHLD CURSP
SHLD SPTEMP
LHLD CURRC
SHLD RCTEMP
LHLD CURAB
SHLD ABTEMP
LHLD CURTH
SHLD SHTIME

CALL UPDT2
POP 1 ;AB VALUE
POP 5 ;RC VALUE

; SET THE APPROPRIATE BIT ON RC VALUE TO INDICATE
; A MAXIMUM BRTN
MOV 1,0
ORI 10H ;FIRST 4 BITS SHOULD = 00012
MOV 1,A

POP 1 ;SM VALUE

```

```

LHLD SHTIME
SHLD TVALUE
LHLD CURSP
SHLD SPTEMP
LHLD CURRC
SHLD RCTEMP
LHLD CURAB
SHLD ABTEMP
LHLD CURTH
SHLD SHTIME

1. STORE RC COUNTS
2. RC COUNTS <= RCTOINT
3. RCTOINT <= 0
4. STORE AB COUNTS
5. AB COUNTS <= ABTOINT
6. ABTOINT <= 0
CALL UPDT3

POP 8 ;AB VALUE
POP 0 ;RC VALUE

; SET THE APPROPRIATE BIT ON RC VALUE TO INDICATE
; A MINIMUM BRTN
MOV A,0
ORI 0FH ;FIRST 4 BITS SHOULD = 00012
MOV 0,A

POP H ;SM VALUE
STC ;SET CARRY
RET

ELSE
;NOT A LARGE ENOUGH SLOPE
11. ADD RC DERIV TO RCTOINT
12. ADD AB DERIV TO ABTOINT
CALL UPDT3

ORA A
RET
ENDIF
ELSE
LHLD SPTEMP
LOED CURSP
XCHG A
ORA A
DSCC 0

STC
RET
ELSE
;NOT LARGE ENOUGH SLOPE
CALL UPDT3
ORA A
RET
ENDIF
ENDIF

UPDT1:
11. ADD CURRENT ABS(RC DERIV) PT TO RC COUNTS
12. ADD RCTOINT TO RC COUNTS
13. RCTOINT <= 0
14. ADD CURRENT ABS(AB DERIV) TO AB COUNTS
15. ADD ABTOINT TO AB COUNTS
16. ABTOINT <= 0

LHLD ABSRCO
LDA RCTINPHS
IF (.A,IS,ZERO) THEN RC : IN PHASE WITH SUM
LOED RCIN
DAO 0
SHLD RCIN

ELSE ;RC OUT OF PHASE WITH SUM
LOED RCOUT
DAO 0
SHLD RCOUT

ENDIF

LHLD RCIN
LOED RCTOINT
DAO 0
SHLD RCIN

LHLD RCOUT
LOED RCTOINT
DAO 0
SHLD RCOUT

LXI H,0
SHLD RCTOINT
SHLD RCTOINT

LHLD ABSABO
LDA ABINPHS
IF (.A,IS,ZERO) THEN AB : IN PHASE WITH SUM

```

```

ELSE          IAB OUT OF PHASE WITH SUM
    LOED ABOUT
    DAD 0
    SHLD ABOUT
ENDIF
LHLD ABIN
LOED ABTOUT
DAD 0
SHLD ABIN
LHLD ABOUT
LOED ABTOUT
DAD 0
SHLD ABOUT
LXI H,0
SHLD ABTOUT
SHLD ABIN
SUM: DO NOT SEPERATE INPHASE AND OUT OF PHASE C-LES
    ADD THEM ALL
    LHLD ABSEMO
    LOED SHIO
    DAD 0
    SHLD SHIO
    LHLD SHIO
    LOED SHTOUT
    DAD 0
    SHLD SHTOUT
    LXI H,0
    SHLD SHTOUT
    RET
UPDT2:
1. STORE RC COUNTS
2. RC COUNTS (= RCTOUT
3. RCTOUT (= 0
4. STORE AB COUNTS
5. AB COUNTS (= ABTOUT
6. ABTOUT (= 0

```

```

LDA RCINPHS
IF (A,15,11ZER0) THEN INPHASE
    LHLD ABSRCO
    LOED RCTOUT
    DAD 0
    SHLD RCTOUT
ELSE
    LHLD ABSRCO
    LOED RCTOUT
    DAD 0
    SHLD RCTOUT
ENDIF
LDA ABINPHS
IF (A,15,11ZER0) THEN INPHASE
    LHLD ABSABO
    LOED ABTOUT
    DAD 0
    SHLD ABTOUT
ELSE
    LHLD ABSABO
    LOED ABTOUT
    DAD 0
    SHLD ABTOUT
ENDIF
SUM: DO NOT SEPERATE INPHASE AND OUT OF PHASE
    LHLD ABSEMO
    LOED SHTOUT
    DAD 0
    SHLD SHTOUT
    RET
GTSIGN:
    LHLD RCSERV ;CURRENT RC DERIV VALUE
    MOV A,H
    ANI 80H
    STA RCSEGN
    LHLD ABSERV ;CURRENT AB DERIV VALUE
    MOV A,H
    ANI 80H
    STA ABSEGN
    LHLD SHDERV

```

```

LHLD RCOUT
SHLD RCOUTUOL
LHLD RCTOUT
SHLD RCIN
LHLD RCTOUT
SHLD RCOUT
LXI H,0
SHLD RCTOUT
SHLD RCTOUT
LHLD ABIN
SHLD ABOUTUOL
LHLD ABOUT
SHLD ABOUTUOL
LHLD ABTOUT
SHLD ABIN
LHLD ABTOUT
SHLD ABOUT
LXI H,0
SHLD ABTOUT
SHLD ABTOUT
SUM: DO NOT SEPERATE INPHASE AND OUT OF PHASE
    LHLD SHIO
    SHLD SHVOL
    LHLD SHTOUT
    SHLD SHIO
    LXI H,0
    SHLD SHTOUT
    RET
UPDT3:
1. ADD ABS(RCSERV) TO RCTOUT
2. ADD ABS(ABSERV) TO ABTOUT

```

```

MOV A,H
ANI 80H
STA SHSIGN
MOV B,A
LDA RCSEGN
IF (A,15,11ZER0) THEN RC INPHASE W/ SUM
    XRA A
    CMA
    STA RCINPHS
ELSE
    XRA A
    STA RCINPHS
ENDIF
LDA SHSIGN
MOV B,A
LDA ABSEGN
IF (A,15,11ZER0) THEN AB INPHASE W/ SUM
    XRA A
    CMA
    STA ABINPHS
ELSE
    XRA A
    STA ABINPHS
ENDIF
RET
DERIVS:
;*****
;DERIV USING 9 POINTS
;*****
;DERIVATIVE USING 3 POINTS
;SACOFST HOLDS OFFSET TO GET Y3 FROM RAW DATA BUFFERS
;ASSUME: RCY1,RCY2,RCY3,RCY4, ABY1,ABY2,ABY3,ABY4 ALREADY SET
    LXI 0,RCOUFF
    LHLD SACOFST
    DAD 0
    MOV E,H
    MOV H

```



```

1 SEE IF AT LEAST 10 POINTS IN RING BUFFER
1

```

```

1 LXLD RIPT
1 LXI D,20
1 ORA A
1 ORC D
1 IF (PSU,IS,CARRY) THEN ( 10 PTS
1 STC
1 ELSE
1

```

```

1
1 SET UP INIT VALUES IN RCY1,RCY2,RCY3,RCY4
1 ABY1,ABY2,ABY3,ABY4,AND SRCOFFST
1

```

```

1 LXI H,RCBUFF
1 MOV E,H
1 INX H
1 MOV D,H
1 SDEO RCY1
1

```

```

1 INX H
1 MOV E,H
1 INX H
1 MOV D,H
1 SDEO RCY2
1

```

```

1 INX H
1 MOV E,H
1 INX H
1 MOV D,H
1 SDEO RCY3
1

```

```

1 INX H
1 MOV E,H
1 INX H
1 MOV D,H
1 SDEO RCY4
1

```

```

1 INX H
1 MOV E,H
1 INX H
1 MOV D,H
1 SDEO RCY5
1

```

```

1 INX H
1 MOV E,H
1 INX H
1 MOV D,H
1 SDEO RCY6
1

```

```

1 SDEO ABY7
1
1 INX H
1 MOV E,H
1 INX H
1 MOV D,H
1 SDEO ABY8
1

```

```

1 LXI H,14 ; OFFSET FOR THE 1ST VALUE IN BUFFER
1 SLD SRCOFFST
1

```

```

1 LXI H,2 ; OFFSET IF "HOLE" IS
1 SLD RST
1

```

```

1 LXI H,0
1 SLD INTCPS
1 ORA A
1 ENOIF
1 RET
1

```

```

1 .END
1

```

```

1 INX H
1 MOV D,H
1 SDEO RCY7
1
1 INX H
1 MOV E,H
1 INX H
1 MOV D,H
1 SDEO RCY8
1

```

```

1 LXI H,ABCBUFF
1 MOV E,H
1 INX H
1 MOV D,H
1 SDEO ABY1
1

```

```

1 INX H
1 MOV E,H
1 INX H
1 MOV D,H
1 SDEO ABY2
1

```

```

1 INX H
1 MOV E,H
1 INX H
1 MOV D,H
1 SDEO ABY3
1

```

```

1 INX H
1 MOV E,H
1 INX H
1 MOV D,H
1 SDEO ABY4
1

```

```

1 INX H
1 MOV E,H
1 INX H
1 MOV D,H
1 SDEO ABY5
1

```

```

1 INX H
1 MOV E,H
1 INX H
1 MOV D,H
1 SDEO ABY6
1

```

```

1 INX H
1 MOV E,H
1 INX H
1 MOV D,H
1

```

```

1 IDENT PRIMAL
1 INSERT FMAC,SRC
1 INSERT SP90,ASH
1 INSERT CALCOM
1 INSERT LP2COM
1 INSERT STCOM
1

```

```

1 .EXTERNAL PPTIME
1

```

```

1 .EXTERNAL LPON,LPOFF
1 FINIT
1 .EXTERNAL POLIO
1 .ENTRY PRIMAL
1

```

```

1 THIS ROUTINE PRINTS OUT THE DELTA VALUES:
1

```

```

1 SUM SP SUM/SP
1

```

```

1 THERE ARE (1-MAXINT) - 1 VALUES
1

```

```

1 PRIMAL:
1

```

```

1 ASSUME VALUE OF PPT IS IN -PES ON ENTRY TO THIS ROUTINE
1

```

```

1 PPTA PPT
1 ONLY MAKE HARD COPY
1 GO:
1

```

```

1 LXLD MAXINT
1 ORC H
1 SLAR L
1 PALR H
1 SLD CMT
1 LDI 4
1 LDI H,VALUES
1 SLD XIPTR
1 LXI H,0
1 SLD CPTTR ; OFFSET FOR SP,RC,AB
1 XRA A
1 STA PTF ; THIS IS ZERO WHEN
1 ; CDS & TIME THRU LOOP
1

```

```

1 XRA A
1 CMA
1 STA HROFLO
1

```

```

1 MVI A,10
1 STA LOHT
1 ONLY MAKE HARD COPY
1 CALL LPON
1

```



```

      .RELOC
      .END

```

CLXIK.

IF (PSW,IS,ZERO) THEN 1 MINUTE

LDA BAG

```

LXI    D,ABUFF
CALL   PRNG

```

LHLD	SPAD	SPIROMETER VALUE
1	10	10
2	10	10
3	10	10
4	10	10
5	10	10
6	10	10
7	10	10
8	10	10
9	10	10
10	10	10
11	10	10
12	10	10
13	10	10
14	10	10
15	10	10
16	10	10
17	10	10
18	10	10
19	10	10
20	10	10
21	10	10
22	10	10
23	10	10
24	10	10
25	10	10
26	10	10
27	10	10
28	10	10
29	10	10
30	10	10
31	10	10
32	10	10
33	10	10
34	10	10
35	10	10
36	10	10
37	10	10
38	10	10
39	10	10
40	10	10
41	10	10
42	10	10
43	10	10
44	10	10
45	10	10
46	10	10
47	10	10
48	10	10
49	10	10
50	10	10
51	10	10
52	10	10
53	10	10
54	10	10
55	10	10
56	10	10
57	10	10
58	10	10
59	10	10
60	10	10
61	10	10
62	10	10
63	10	10
64	10	10
65	10	10
66	10	10
67	10	10
68	10	10
69	10	10
70	10	10
71	10	10
72	10	10
73	10	10
74	10	10
75	10	10
76	10	10
77	10	10
78	10	10
79	10	10
80	10	10
81	10	10
82	10	10
83	10	10
84	10	10
85	10	10
86	10	10
87	10	10
88	10	10
89	10	10
90	10	10
91	10	10
92	10	10
93	10	10
94	10	10
95	10	10
96	10	10
97	10	10
98	10	10
99	10	10
100	10	10

DIVIDE BY 8 (SIGNED SHIFT RIGHT)
CHANGE TO SHIFT ONLY 2 TIMES
BYTES PER INSTRUCTION

1ADD OFFST SO ALL 1 > 0

LNLD	SMAD	SUM VALUE
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7
8	8	8
9	9	9
10	10	10
11	11	11
12	12	12
13	13	13
14	14	14
15	15	15
16	16	16
17	17	17
18	18	18
19	19	19
20	20	20
21	21	21
22	22	22
23	23	23
24	24	24
25	25	25
26	26	26
27	27	27
28	28	28
29	29	29
30	30	30
31	31	31
32	32	32
33	33	33
34	34	34
35	35	35
36	36	36
37	37	37
38	38	38
39	39	39
40	40	40
41	41	41
42	42	42
43	43	43
44	44	44
45	45	45
46	46	46
47	47	47
48	48	48
49	49	49
50	50	50
51	51	51
52	52	52
53	53	53
54	54	54
55	55	55
56	56	56
57	57	57
58	58	58
59	59	59
60	60	60
61	61	61
62	62	62
63	63	63
64	64	64
65	65	65
66	66	66
67	67	67
68	68	68
69	69	69
70	70	70
71	71	71
72	72	72
73	73	73
74	74	74
75	75	75
76	76	76
77	77	77
78	78	78
79	79	79
80	80	80
81	81	81
82	82	82
83	83	83
84	84	84
85	85	85
86	86	86
87	87	87
88	88	88
89	89	89
90	90	90
91	91	91
92	92	92
93	93	93
94	94	94
95	95	95
96	96	96
97	97	97
98	98	98
99	99	99
100	100	100

1
:FILL ABOVE WITH ZEROS

```

ENDIF
FILL ACBUFF WITH ZEROES
LXI H,0
SHLO DATA
LHLO RIPT
LXI D,ACBUFF
CALL PRNG

INCREMENT RIPT FOR NEXT TIME

LHLO TIME
INX H
SHLO TIME
SHLO DATA
LHLO RIPT
LXI D,THUBFF
CALL PRNG

LHLO RIPT
SHLO DORIPT
LXI B,RSIZE
LHLO RIPT
CALL PTRUPD
SHLO RIPT

LDA WRAPP
IF (A,IS,ZERO) ;SEE IF RIPT RESET TO 0
LXI D,0
LHLO RIPT
CALL TONE
IF (PSM,IS,ZERO)
MVI A,OFFH
STA WRAPP
ENDIF
ENDIF
RET

CHKFLIP:
CHECK IF NEED TO FLIP
LDA FLIPV
IF (A,IS,ZERO)

```

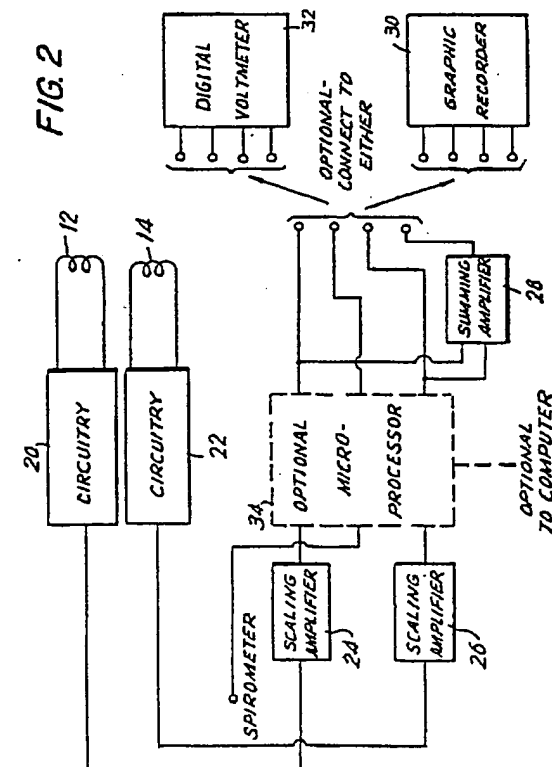
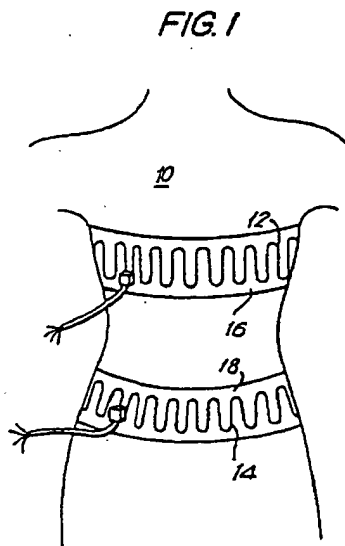


FIG. 1は、双腕にセンサーを装着した人の胸に、ラインが、その時、行われる。そのプログラムは、胸郭部と腹部信号用のデルタ値を自動的に演算し、見当違いの点を除去し、残ったデルタ値の合計から胸郭部と腹部用の平均値を計算し、それらの平均偏差を算出して、方程式Mから比例係数Zを算出する。胸郭部用の縮尺増幅器は、その時、比例係数Zに調整される。胸郭部及び腹部信号の和を差出すスビグラフMからのリアルタイム出力は、その時、上記で十分に説明したように、干渉量に比例する。プログラムはまた、次の5分間隔毎に比例係数Zを再計算する。もし再計算された値が、受け入れられる量以上に1.0から異なる場合には、計算ルーチンはリランすることができ。プログラムはまた、例えば肺活量測定によって引出される実際の干渉量を入力するときに、もし指令されれば方程式Nから縮尺係数Mを計算する。

ここに、この発明の好ましい実施形態とそれに対して示されるある変形が示された記述されているが、この発明の精神と範囲から離れることなく変形や特徴が更になされることは明らかである。従って、前述の説明は実例として解釈されるべきものであり、限定された意味に解釈されるものではなく、この発明の範囲は、次の特許請求の範囲によって限定されるものである。

From 1973-1978, around 1977-1978

特許法第17条第1項又は第17条の2の規定
による補正の掲載

昭和62年特許願第501323号(特表昭63-
502563号、昭和63年9月29日発行公表特許
公報)については特許法第17条第1項又は第17条の2
の規定による補正があったので下記のとおり掲載する。

Int.Cl. ⁴	識別 記号	庁内整理番号
A61B 5/08		7831-4C

補正の内容

- (1)請求の範囲の翻訳文を別紙のとおり補正する。
- (2)明細書の翻訳文第3頁第19行の「分割」を「割算」に訂正する。
- (3)同翻訳文第6頁第9行と第10行との間に下記の記事を挿入する。

「負荷の算定」

この発明は、呼吸器に対する胸廓部信号を示す胸廓部値に相当する信号を提供する手段、呼吸器に対する腹部信号を示す腹部値に相当する信号を提供する手段、および胸廓部信号と腹部信号のうちの少なくとも一方に呼吸器に対する胸廓部と腹部の相対的寄与を反映する所定の重み係数を算出する手段を備えた、呼吸器を非侵襲的に測定する装置を修正するための方法の改良に関する。

重み係数を測定するために改良された校正方法は、未校正の胸廓部信号および腹部信号における各呼吸ごとの相対的な振幅を示す複数のパラメータ値を、実質的に定常状態にある呼吸

手続補正書

平成元年2月10日

特許庁長官 吉田 文 殿

1. 事件の表示
昭和62年特許願第501323号
2. 発明の名称
呼吸モニタ装置用の単一姿勢における非侵襲的校正技法
3. 提出をする者
事件との関係 特許出願人
住 所 米国、フロリダ 33139-1432 マイアミ、
ウェスト・アベニュー 1842
名 称 レスビトレイス・コーポレーション
代表者 ジェクサー・マービン
4. 代理人 〒530
住 所 大阪市北区西天満5丁目1-3 クオーター・ワンビル
電話 (06) 365-0718
氏 名 弁護士(6524) 野 村 浩太
5. 補正命令の日付 自 発
6. 補正の対象
「請求の範囲の翻訳文」および「明細書の翻訳文」
7. 補正の内容
別紙のとおり



方 式 変 更

のベースライン期間にわたって合計し、ついで、胸廓部信号および腹部信号のうちのいずれか一方の値の合計の平均値の平均変動率を、他方の値の合計の平均値の平均変動率で除し、その後、他方の値に、その除算ステップから得られ、かつ、重み係数を算出する値を乗じる操作からなっている。この発明に従って校正するための装置も開示されている。

さらに、この発明の特徴および利点は、以下の詳細な説明およびその図面と実施態様の説明図面から、より充分に明らかにされる。」

- (4)同翻訳文第10頁において、

第23行の「表示する」を「示す」に訂正する。

第24行の「評価」を「理解」に訂正する。

- (5)同翻訳文第12頁において、

第10-11行の「平均変化率を表示する」ということは評価されるべきである。」を「平均変動率を示すということを理解すべきである。」に訂正する。

第16行の「等重処理」を「等寄処理」に訂正

する。

第19～21行の「もちろん……償うものである。」を削除する。

第22行の「しがしながら、」を削除する。

(6)同翻訳文第13頁において、

第2～3行の「見当惑いの点」を「突出点」に訂正する。

第14行の「見当惑いの点」を「突出点」に訂正する。

(7)同翻訳文第14頁において、

第16～17行の「縮尺係数Mによって……増大させることにより、」を「加算増幅器28のゲインに縮尺係数Mを乗じることにより、」に訂正する。

第18行の「出力が半満値を表示する半分の値となる。」を「出力は半満値を半定量的に表わす。」に訂正する。

第19行の「半分の値という用語は、」を「半定量的という用語を用いているのは、」に訂正する。

- 2 -

する。

「ベースライン期間において肺活量計による肺活量測定を行いその示度を読み取っても、この発明に基いた校正技法、すなわち非侵襲的な校正技法の目的を達成することができないことはいうまでもない。」

- 3 -

第20行の「決定され」を「決定された」に訂正する。

第21行の「決定されたために用いられる。」を「搬送されたためである。」に訂正する。

(8)同翻訳文第15頁において、

第5行の「約.7」を「約0.7」に訂正する。

第12行の「一般的に利用される。」を「利用することができる。」に訂正する。

第16～17行の「以下のテーブルA」を「この記述の最後」に訂正する。

第17行の「ジログ」を「ダイログ」に訂正する。

(9)同翻訳文第15頁において、

第4行の「見当惑いの点」を「突出点」に訂正する。

第12行の「以上に」を「以上であり」に訂正する。

第14行の「引出される」を「得られる」に訂正する。

第15行と第16行との間につきの文章を挿入

4.

請求の範囲

1. 呼吸器に対する胸部部番号を示す胸部部番号に
応答する番号を提供する手段、呼吸器に対する
前記胸部部番号を示す胸部部番号に応答する番号を提供する手段、および、前記胸部部番号と胸部部番号のうちの、
少なくとも一方に、呼吸器に対する前記胸部部番号と
胸部の相対的番号を反映する所定の重み係数を乗
じる手段を備えた、該装置の呼吸器を非侵襲的に
測定する装置のための校正技法であって、

(a) 実質的に定常状態にある呼吸のベース
ライン期間にわたる前記胸部部番号分の
デルタ値を合計し、

(b) 実質的に定常状態にある呼吸のベース
ライン期間にわたる前記胸部部番号用の
デルタ値を合計し、

(c) 前記胸部部番号用デルタ値および胸部
部番号用デルタ値のうちのいずれか一方
の合計の平均値の平均変動率を、前記
他方のデルタ値の合計の平均値の平均
変動率によって除し、そして

- 1 -

(d) 前記地方の番号に、ステップ(c)から得られた商に等しい値を係数乗じ

る

ことによって前記重み係数を非線形的に決定する
校正技法。

2. 前記の合計するステップの前に、前記ベースライン期間における胸郭部値およびこれと対応する腹壁値の合計の平均変動率を決定するステップと、合計値が所定合計値の前記平均変動率と異なる胸郭部値用デルタ値と腹壁値用デルタ値を除去するステップとをさらに備えている請求の範囲第1項に記載の技法。

3. 前記一方の値と前記地方の重み付き値とを合計して呼吸量に比例した合計値を提供するステップをさらに備えている請求の範囲第1項に記載の技法。

4. 前記地方の重み付き値を用いて所定範囲でステップ(a)、(b)を繰り返すステップと、前記の商が所定値1.0と異なる場合には指示を与えるステップとをさらに備えている請求の範囲

第1項に記載の技法。

5. 一定時間内ににおける実際の呼吸量を示す値を提供し、ついで、換算係数を求めるために前記実際の呼吸量を前記時間内に得られた合計値で除じ、その商、実際の呼吸量に定量的に関連した値を求めるために前記合計値に前記換算係数を乗じるステップをさらに備えている請求の範囲第3項に記載の技法。

6. 前記地方の重み付き値によって所定時間内における前記の値を再計算する手段と、前記の商が1.0と異なる場合には所定値による指示を与える手段とをさらに備えている請求の範囲第1項に記載の技法。

7. 一定時間内における実際の呼吸量を示す値を提供する手段、換算係数を求めるために前記実際の呼吸量を示す値を前記時間内に得られた前記合計値で除す手段、および実際の呼吸量に定量的に関連した値を得るために前記合計値に前記換算係数を乗じる手段をさらに備えている請求の範囲第3項に記載の技法。

- 2 -

8. 呼吸量に対する胸郭部値を示す胸郭部値に
応答する値を提供する手段、呼吸量に対する腹
壁部値を示す腹壁部値に
応答する値を提供する手段、および、前記胸郭部値と腹壁部値の2
つの少なくとも一方に、呼吸量に対する前記胸郭
部値と腹壁部値の相対的寄与を反映する所定の
重み係数を乗じる手段を備えた、被験者の呼吸量
を非線形的に測定する装置のための校正装置であつ
て、

実質的に定常状態にある呼吸のベースラ
イン期間にわたる前記胸郭部値用および
腹壁部値用のデルタ値を個別に合計する手
段、

前記胸郭部値用デルタ値および腹壁部
値用デルタ値のうちのいずれか一方の合計
の平均値の平均変動率を、前記地方の値の
ための前記デルタ値の合計の平均値の平
均変動率によって除す手段、および、

前記地方の値に前記除算手段によって
得られた商を乗じるために前記換算手段を

提供する手段

を有する校正装置。

9. ベースライン期間にわたる胸郭部値および
これと対応する腹壁部値の合計の平均変動率を決定
する手段と、合計値が所定合計値の前記平均変動
率と異なる胸郭部値用デルタ値および腹壁部
値用デルタ値を除去する手段とをさらに備えてい
る請求の範囲第8項に記載の校正装置。

10. 呼吸量に比例する合計値を提供するため
に前記一方の値と前記地方の重み付き値とを
合計する手段をさらに備えている請求の範囲第8
項に記載の校正装置。

- 3 -